Exascale Computing Project Software Activities

Michael A. Heroux (Sandia National Labs)
Director of Software Technology, Exascale Computing Project

HPC User Forum
Santa Fe, NM
April 2, 2019
The Exascale Computing Project (ECP) enables US revolutions in technology development; scientific discovery; healthcare; energy, economic, and national security.

**ECP mission**

- **Develop exascale-ready applications** and solutions that address currently intractable problems of strategic importance and national interest.

- **Create and deploy an expanded and vertically integrated software stack** on DOE HPC exascale and pre-exascale systems, defining the enduring US exascale ecosystem.

- **Deliver US HPC vendor technology advances and deploy ECP products** to DOE HPC pre-exascale and exascale systems.

**ECP vision**

- **Deliver exascale simulation and data science innovations and solutions to national problems** that enhance US economic competitiveness, change our quality of life, and strengthen our national security.
The ECP is part of the broader DOE Exascale Computing Initiative (ECI)

Exascale Computing Project (ECP)

Selected program office application development (BER, BES, NNSA)

Exascale system procurement projects & facilities
- ALCF-3 (Aurora)
- OLCF-5 (Frontier)
- ASC ATS-4 (El Capitan)

ECI partners
- US DOE Office of Science (SC) and National Nuclear Security Administration (NNSA)

ECI mission
- Accelerate R&D, acquisition, and deployment to deliver exascale computing capability to DOE national labs by the early- to mid-2020s

ECI focus
- Delivery of an enduring and capable exascale computing capability for use by a wide range of applications of importance to DOE and the US

Three Major Components of the ECI
The three technical areas in ECP have the necessary components to address these challenges and meet national goals.

**Performant mission and science applications @ scale**

- **Foster application development**
- **Ease of use**
- **Diverse architectures**
- **HPC leadership**

**Application Development (AD)**
- Develop and enhance the predictive capability of applications critical to the DOE

**Software Technology (ST)**
- Produce expanded and vertically integrated software stack to achieve full potential of exascale computing

**Hardware and Integration (HI)**
- Integrated delivery of ECP products on targeted systems at leading DOE computing facilities

25 applications ranging from national security, to energy, earth systems, economic security, materials, and data

80+ unique software products spanning programming models and run times, math libraries, data and visualization

6 vendors supported by PathForward focused on memory, node, connectivity advancements; deployment to facilities
The three technical areas in ECP have the necessary components to address these challenges and meet national goals.

**Performant mission and science applications @ scale**

<table>
<thead>
<tr>
<th>Foster application development</th>
<th>Ease of use</th>
<th>Diverse architectures</th>
<th>HPC leadership</th>
</tr>
</thead>
</table>

**Application Development (AD)**
Develop and enhance the predictive capability of applications critical to the DOE

**Software Technology (ST)**
Produce expanded and vertically integrated software stack to achieve full potential of exascale computing

**Hardware and Integration (HI)**
Integrated delivery of ECP products on targeted systems at leading DOE computing facilities

25 applications ranging from national security, to energy, earth systems, economic security, materials, and data

80+ unique software products spanning programming models and run times, math libraries, data and visualization

6 vendors supported by PathForward focused on memory, node, connectivity advancements; deployment to facilities
ECP’s 25 applications target national problems in DOE mission areas

<table>
<thead>
<tr>
<th>National security</th>
<th>Energy security</th>
<th>Economic security</th>
<th>Scientific discovery</th>
<th>Earth system</th>
<th>Health care</th>
</tr>
</thead>
<tbody>
<tr>
<td>Next-generation, stockpile stewardship codes</td>
<td>Turbine wind plant efficiency</td>
<td>Additive manufacturing of qualifiable metal parts</td>
<td>Cosmological probe of the standard model of particle physics</td>
<td>Accurate regional impact assessments in Earth system models</td>
<td>Accelerate and translate cancer research (partnership with NIH)</td>
</tr>
<tr>
<td>Reentry-vehicle-environment simulation</td>
<td>Design and commercialization of SMRs</td>
<td>Urban planning</td>
<td>Validate fundamental laws of nature</td>
<td>Stress-resistant crop analysis and catalytic conversion of biomass-derived alcohols</td>
<td></td>
</tr>
<tr>
<td>Multi-physics science simulations of high-energy density physics conditions</td>
<td>Nuclear fission and fusion reactor materials design</td>
<td>Reliable and efficient planning of the power grid</td>
<td>Plasma wakefield accelerator design</td>
<td>Metagenomics for analysis of biogeochemical cycles, climate change, environmental remediation</td>
<td></td>
</tr>
<tr>
<td></td>
<td>Subsurface use for carbon capture, petroleum extraction, waste disposal</td>
<td>Seismic hazard risk assessment</td>
<td>Light source-enabled analysis of protein and molecular structure and design</td>
<td></td>
<td></td>
</tr>
<tr>
<td></td>
<td>High-efficiency, low-emission combustion engine and gas turbine design</td>
<td></td>
<td>Find, predict, and control materials and properties</td>
<td></td>
<td></td>
</tr>
<tr>
<td></td>
<td>Scale up of clean fossil fuel combustion</td>
<td></td>
<td>Predict and control stable ITER operational performance</td>
<td></td>
<td></td>
</tr>
<tr>
<td></td>
<td>Biofuel catalyst design</td>
<td></td>
<td>Demystify origin of chemical elements</td>
<td></td>
<td></td>
</tr>
</tbody>
</table>

ECP's 25 applications target national problems in DOE mission areas.
Common R&D activities/challenges that applications face

1) Porting to accelerator-based architectures
2) Exposing additional parallelism
3) Coupling codes to create new multiphysics capability
4) Adopting new mathematical approaches
5) Algorithmic or model improvements
6) Leveraging optimized libraries
The three technical areas in ECP have the necessary components to address these challenges and meet national goals.

**Performant mission and science applications @ scale**

<table>
<thead>
<tr>
<th>Foster application development</th>
<th>Ease of use</th>
<th>Diverse architectures</th>
<th>HPC leadership</th>
</tr>
</thead>
</table>

**Application Development (AD)**
- Develop and enhance the predictive capability of applications critical to the DOE

**Software Technology (ST)**
- Produce expanded and vertically integrated software stack to achieve full potential of exascale computing

**Hardware and Integration (HI)**
- Integrated delivery of ECP products on targeted systems at leading DOE computing facilities

- **25 applications** ranging from national security, to energy, earth systems, economic security, materials, and data
- **80+ unique software products** spanning programming models and run times, math libraries, data and visualization
- **6 vendors** supported by PathForward focused on memory, node, connectivity advancements; deployment to facilities
Department of Energy (DOE) Roadmap to Exascale Systems

An impressive, productive lineup of *accelerated node* systems supporting DOE’s mission

<table>
<thead>
<tr>
<th>Pre-Exascale Systems [Aggregate Linpack (Rmax) = 323 PF]</th>
<th>First U.S. Exascale Systems</th>
</tr>
</thead>
<tbody>
<tr>
<td><strong>2012</strong></td>
<td><strong>2021-2023</strong></td>
</tr>
<tr>
<td>Titan (9) ORNL Cray/AMD/NVIDIA</td>
<td>ORNL TBD</td>
</tr>
<tr>
<td>Mira (24) ANL IBM BG/Q</td>
<td>Aurora ANL Intel/Cray</td>
</tr>
<tr>
<td>Theta (24) ANL Cray/Intel KNL</td>
<td>LBNL Cray/Intel Xeon/KNL</td>
</tr>
<tr>
<td>Cori (12) LBNL Cray/Intel Xeon/KNL</td>
<td>Perlmutter LBNL Cray/AMD/NVIDIA</td>
</tr>
<tr>
<td>Sequoia (10) LLNL IBM BG/Q</td>
<td>Sierra (2) LLNL IBM/NVIDIA</td>
</tr>
<tr>
<td>Trinity (8) LANL/SNL Cray/Intel Xeon/KNL</td>
<td>LANL/SNL TBD</td>
</tr>
</tbody>
</table>

Sierra (2) LLNL IBM/NVIDIA

First U.S. Exascale Systems

- **2021-2023**
  - ORNL TBD
  - Aurora ANL Intel/Cray
  - LBNL Cray/Intel Xeon/KNL
  - Perlmutter LBNL Cray/AMD/NVIDIA
  - Sierra (2) LLNL IBM/NVIDIA
  - LANL/SNL TBD
Hardware and Integration is designed to enable integration of ECP’s products into HPC environments at the Facilities. ECP will meet its objectives on Facility resources.

- **Applications**
- **Software SDKs and stack components**
- **Early hardware R&D**

**Hardware and Integration Elements**

- Facility resource utilization
- Developer training and productivity
- PathForward

**Facilities are our customers**

- DOE HPC Facilities
- US vendor system offerings
The three technical areas in ECP have the necessary components to address these challenges and meet national goals.

**Performant mission and science applications @ scale**

- **Foster application development**
- **Ease of use**
- **Diverse architectures**
- **HPC leadership**

**Application Development (AD)**
- Develop and enhance the predictive capability of applications critical to the DOE

**Software Technology (ST)**
- Produce expanded and vertically integrated software stack to achieve full potential of exascale computing

**Hardware and Integration (HI)**
- Integrated delivery of ECP products on targeted systems at leading DOE computing facilities

- **25 applications** ranging from national security, to energy, earth systems, economic security, materials, and data
- **80+ unique software products** spanning programming models and run times, math libraries, data and visualization
- **6 vendors** supported by PathForward focused on memory, node, connectivity advancements; deployment to facilities
ECP Software: productive, sustainable ecosystem

Goal
Build a comprehensive, coherent software stack that enables application developers to productively write highly parallel applications that effectively target diverse exascale architectures

- Extend current technologies to exascale where possible
- Perform R&D required for new approaches when necessary
- Guide, and complement, and integrate with vendor efforts
- Develop and deploy high-quality and robust software products
The Bottom Line for ECP Software Technology

• Next-generation **HPC technologies for 90 open source scientific software products**

• The performance potential of leadership computers in preparation for exascale

• **Software development kits (SDKs)** with turnkey installation and interoperability

• The **Extreme-scale Scientific Software Stack (E4S):**
  - Target: Comprehensive software environment for HPC scientific applications
  - Tested on growing collection of HPC platforms in preparation for Exascale systems
  - Managed complexity using SDKs as components
  - From-source builds for leadership environments
  - Pre-built containers for development, debugging and portability

• A commitment to software quality leveraging industry best practices

• A legacy to build upon for US security, science, industry and technology leadership
ECP ST Software Ecosystem

Collaborators (with ECP HI)

ECP Applications
Facilities
Vendors
HPC Community

Software Ecosystem & Delivery

ECP Software Technology

Programming Models
Runtimes

Development Tools

Mathematical Libraries

Data & Visualization
We work on products applications need now and into the future

Key themes:
- Exploration/development of new algorithms/software for emerging HPC capabilities:
- High-concurrency node architectures and advanced memory & storage technologies.
- Enabling access and use via standard APIs.

Software categories:
- The next generation of well-known and widely used HPC products (e.g., MPICH, OpenMPI, PETSc)
- Some lesser used but known products that address key new requirements (e.g., Kokkos, RAJA, Spack)
- New products that enable exploration of emerging HPC requirements (e.g., SICM, zfp, UnifyCR)

<table>
<thead>
<tr>
<th>Example Products</th>
<th>Engagement</th>
</tr>
</thead>
<tbody>
<tr>
<td>MPI – Backbone of HPC apps</td>
<td>Explore/develop MPICH and OpenMPI new features &amp; standards.</td>
</tr>
<tr>
<td>OpenMP/OpenACC –On-node parallelism</td>
<td>Explore/develop new features and standards.</td>
</tr>
<tr>
<td>Performance Portability Libraries</td>
<td>Lightweight APIs for compile-time polymorphisms.</td>
</tr>
<tr>
<td>LLVM/Vendor compilers</td>
<td>Injecting HPC features, testing/feedback to vendors.</td>
</tr>
<tr>
<td>Perf Tools - PAPI, TAU, HPCToolkit</td>
<td>Explore/develop new features.</td>
</tr>
<tr>
<td>Math Libraries: BLAS, sparse solvers, etc.</td>
<td>Scalable algorithms and software, critical enabling technologies.</td>
</tr>
<tr>
<td>IO: HDF5, MPI-IO, ADIOS</td>
<td>Standard and next-gen IO, leveraging non-volatile storage.</td>
</tr>
<tr>
<td>Viz/Data Analysis</td>
<td>ParaView-related product development, node concurrency.</td>
</tr>
</tbody>
</table>
ECP software technologies are a fundamental underpinning in delivering on DOE’s exascale mission

- **Programming Models & Runtimes**
  - Enhance and get ready for exascale the widely used MPI and OpenMP programming models (hybrid programming models, deep memory copies)
  - Development of
- **Development Tools**
  - Continued, multifaceted capabilities in portable, open-source LLVM compiler ecosystem to support expected ECP architectures, including support for
- **Math Libraries**
  - Linear algebra, iterative linear solvers, direct linear solvers, integrators and nonlinear solvers, optimization, FFTs, etc
  - Performance on new node architectures;
- **Data and Visualization**
  - I/O via the HDF5 API
  - Insightful, memory-efficient in-situ visualization and analysis – Data reduction via scientific data compression
  - Checkpoint restart
- **Software Ecosystem**
  - Develop features in Spack necessary to support all ST products in E4S, and the AD projects that adopt it
  - Development of Spack stacks for reproducible turnkey deployment of large collections of software
  - Optimization and

- **SOLLVE**
  - New release includes declare mapper for accelerator deep copy
  - Pragmas to direct advanced loop transformations
  - Validation suite for OpenMP for vendor use
- **Exa-PAPI**
  - Performance counters for advanced ECP hardware
  - Software defined events from ECP software stack: co-design new standard API, implement support infrastructure
- **STRUMPACK**
  - Better OpenMP support for HSS
  - Use OpenMP tasking parallelism, better threading for element extraction, sparse-matrix randomized sampling
  - Results in speedups for 2-5X
- **ZFP compression**
  - More accurate than IEEE for given storage cost
  - Same accuracy for half the storage
- **Spack**
  - Spack Stacks for collections of software
  - Spack used by nearly 40 ST products for E4S deployment
ECP ST staff contribute to ISO and *de facto* standards groups: assuring sustainability through standards

<table>
<thead>
<tr>
<th>Standards Effort</th>
<th>ECP ST Participants</th>
</tr>
</thead>
<tbody>
<tr>
<td>MPI Forum</td>
<td>15</td>
</tr>
<tr>
<td>OpenMP</td>
<td>15</td>
</tr>
<tr>
<td>BLAS</td>
<td>6</td>
</tr>
<tr>
<td>C++</td>
<td>4</td>
</tr>
<tr>
<td>Fortran</td>
<td>4</td>
</tr>
<tr>
<td>OpenACC</td>
<td>3</td>
</tr>
<tr>
<td>LLVM</td>
<td>2</td>
</tr>
<tr>
<td>PowerAPI</td>
<td>1</td>
</tr>
<tr>
<td>VTK ARB</td>
<td>1</td>
</tr>
</tbody>
</table>

- **MPI/OpenMP**: Several key leadership positions
- **Heavy involvement in all aspects.**
- **C++**: Getting HPC requirements considered, contributing working code.
- **Fortran**: Flang front end for LLVM.
- **De facto**: Specific HPC efforts.
- **ARB**: Good model for SDKs.
  *Architecture Review Board*
Many ECP ST products are available for broad community use.

For example...

The exascale software ecosystem will be comprised of a wide array of software, all of which are expected to be used by DOE applications; a key ST effort is focused on developing turn-key installations for DOE Facilities through software development toolkits and the Extreme Scale Scientific Software Stack (E4S).
Software Development Kit Motivation

• The exascale software ecosystem will be comprised of a wide array of software, all of which are expected to be used by DOE applications.

• The software must be:
  – interoperable
  – sustainable
  – maintainable
  – adaptable
  – portable
  – scalable
  – deployed at DOE computing facilities

• Provides intermediate coordination points to better manage complexity

• Without these qualities:
  – Value will be diminished
  – Scientific productivity will suffer
Software Development Kits are a key delivery vehicle for ECP

• A collection of related software products (called packages) where coordination across package teams will improve usability and practices and foster community growth among teams that develop similar and complementary capabilities

• Attributes
  – Domain scope: Collection makes functional sense
  – Interaction model: How packages interact; compatible, complementary, interoperable
  – **Community policies**: Value statements; serve as criteria for membership
  – Meta-infrastructure: Encapsulates, invokes build of all packages (Spack), shared test suites
  – Coordinated plans: Inter-package planning. Does not replace autonomous package planning
  – Community outreach: Coordinated, combined tutorials, documentation, best practices

• Overarching goal: Unity in essentials, otherwise diversity
SDK “Horizontal” Grouping:
Key Quality Improvement Driver

Horizontal (vs Vertical) Coupling
– Common substrate
– Similar function and purpose
  • e.g., compiler frameworks, math libraries
– Potential benefit from common Community Policies
  • Best practices in software design and development and customer support
– Used together, but not in the long vertical dependency chain sense
– Support for (and design of) common interfaces
  • Commonly an aspiration, not yet reality
ECP ST SDK community policies: Important team building, quality improvement, membership criteria.

**SDK Community Policy Strategy**

- Review and revise xSDK community policies and categorize
  - Generally applicable
  - In what context the policy is applicable
- Allow each SDK latitude in customizing appropriate community policies
- Establish baseline policies in FY19 Q2, continually refine

**xSDK compatible package**: Must satisfy mandatory xSDK policies:

- M1. Support xSDK community GNU Autoconf or CMake options.
- M2. Provide a comprehensive test suite.
- M3. Employ user-provided MPI communicator.
- M4. Give best effort at portability to key architectures.
- M5. Provide a documented, reliable way to contact the development team.

**Recommended policies**: encouraged, not required:

- R1. Have a public repository.
- R2. Possible to run test suite under valgrind in order to test for memory corruption issues.
- R3. Adopt and document consistent system for error conditions/exceptions.
- R4. Free all system resources it has acquired as soon as they are no longer needed.
- R5. Provide a mechanism to export ordered list of library dependencies.

**xSDK member package**: An xSDK-compatible package, that uses or can be used by another package in the xSDK, and the connecting interface is regularly tested for regressions.

**Prior to defining and complying with these policies, a user could not correctly, much less easily, build hypre, PETSc, SuperLU and Trilinos in a single executable: a basic requirement for some ECP app multi-scale/multi-physics efforts.**

Initially the xSDK team did not have sufficient common understanding to jointly define community policies.
xSDK-0.3.0: Dec 2017… (that was then..)

Multiphysics Application C

Notation: A \rightarrow B:
A can use B to provide functionality on behalf of A

Domain components
- Reacting flow, etc.
- Reusable.

Libraries
- Solvers, etc.
- Interoperable.

Frameworks & tools
- Doc generators.
- Test, build framework.

SW engineering
- Productivity tools.
- Models, processes.

Extreme-Scale Scientific Software Development Kit (xSDK)

https://xsdk.info

xSDK functionality, Dec 2017

Tested on key machines at ALCF, NERSC, OLCF, also Linux, Mac OS X

Alquimia
PFLOTRAN

PETSc
hypre
SUNDIALS

SuperLU
Trilinos

MFEM
More contributed libraries

MAGMA

Spack

HDF5
BLAS

More external software

July 2018: Revisions of xSDK Community Policies
https://xsdk.info/policies

More domain components

PFLOTRAN

Alquimia

PFLOTRAN

MAGMA

Spack

HDF5
BLAS

More external software

More domain components

https://xsdk.info

ALCF, NERSC, OLCF, also Linux, Mac OS X

https://xsdk.info/policies
xSDK Version 0.4.0: December 2018 (this is now)

https://xsdk.info

Each xSDK member package uses or can be used with one or more xSDK packages, and the connecting interface is regularly tested for regressions.

Multiphysics Application C

Application A

Application B

xSDK functionality, Dec 2018

Tested on key machines at ALCF, NERSC, OLCF, also Linux, Mac OS X

Impact: Improved code quality, usability, access, sustainability

Foundation for work on performance portability, deeper levels of package interoperability

Alquimia

SLEPc

hypre

PETSc

SUNDIALS

AMReX

Omega_h

deal.II

PFLOTRAN

PETSc

More domain components

SuperLU

Trilinos

MFEM

Tasmanian

DTK

MAGMA

PHIST

PLASMA

HDF5

BLAS

More external software

STRUMPACK

SLEPc

More libraries

MAGMA

More libraries

Spack

December 2018

- 17 math libraries
- 2 domain components
- 16 mandatory xSDK community policies
- Spack xSDK installer

Domain components
- Reacting flow, etc.
- Reusable.

Libraries
- Solvers, etc.
- Interoperable.

Frameworks & tools
- Doc generators.
- Test, build framework.

SW engineering
- Productivity tools.
- Models, processes.

Extreme-Scale Scientific Software Development Kit (xSDK)
The planned ECP ST SDKs will span all technology areas.
SDK Summary

• SDKs will help reduce complexity of delivery:
  – Hierarchical build targets.
  – Distribution of software integration responsibilities.

• New Effort: Started in April 2018, fully established in August 2018.

• Extending the SDK approach to all ECP ST domains.
  – SDKs create a horizontal coupling of software products, teams.
  – Create opportunities for better, faster, cheaper – pick all three.

• First concrete effort: Spack target to build all packages in an SDK.
  – Decide on good groupings.
  – Not necessarily trivial: Version compatibility issues, Coordination of common dependencies.

• Longer term:
  – Establish community policies, enhance best practices sharing.
  – Provide a mechanism for shared infrastructure, testing, training, etc.
  – Enable community expansion beyond ECP.
Extreme-Scale Scientific Software Stack – E4S

- **E4S**: A Spack-based distribution of ECP ST and related and dependent software tested for interoperability and portability to multiple architectures
- Provides distinction between SDK usability / general quality / community and deployment / testing goals
- Will leverage and enhance SDK interoperability thrust
- Oct: E4S 0.1 - 24 full, 24 partial release products
- Jan: E4S 0.2 - 37 full, 10 partial release products
- Current primary focus: Facilities deployment

Lead: Sameer Shende (U Oregon)
E4S Full Release and Installed Packages

- Adios
- Bolt
- Caliper
- Darshan
- Gasnet
- GEOPM
- GlobalArrays
- Gotcha
- HDF5
- HPCToolkit
- Hypre
- Jupyter
- Kokkos
- Legion
- Libquo
- Magma
- MFEM
- MPICH
- OpenMPI
- PAPI
- Papyrus
- Parallel netCDF
- ParaView
- PETSc/TAO
- Program Database Toolkit (PDT)
- Qthreads
- Raja
- SCR
- Spack
- Strumpack
- Sundials
- SuperLU
- Swift/T
- SZ
- Tasmanian
- TAU
- Trilinos
- VTKm
- Umpire
- UnifyCR
- Veloc
- xSDK
- Zfp
Spack
A flexible package manager for HPC

• Inspired by Homebrew, Nix, some others
• Support scientific stacks with multiple languages
• Flexibility:
  – Build packages many different ways
  – Change compilers and flags in builds
  – Swap implementations of libraries (MPI, BLAS, etc.)
• Run on laptops, Linux clusters, and the largest supercomputers in the world

Easy installation
$ git clone https://github.com/spack/spack
$. spack/share/spack/setup-env.sh
$ spack install hdf5

Easy customization
$ spack install mpileaks@3.3
$ spack install mpileaks@3.3 %gcc@4.7.3 +threads
$ spack install mpileaks@3.3 cppflags="-O3 -g3"
$ spack install mpileaks@3.3 target=haswell
$ spack install mpileaks@3.3 ^mpich@3.2

Over 150,000 downloads in the past year
Over 2,800 software packages
Over 300 contributors from labs, academia, industry

Spack is used worldwide!
Detailed Information about the software technology projects is available in the ECP ST Capability Assessment Report

- Products discussed here are presented with more detail and further citations.
- We classify ECP ST Products deployment as Broad, Moderate or Experimental.
  - Broad and Moderate Deployment is typical suitable for collaboration.
  - Web links are available for almost all products.
  - About 1/3 of ECP ST Products are available as part of the Extreme-scale Scientific Software Stack (E4S) http://e4s.io.

ECP ST Technologies that may be particularly suited to industry interactions

**Programming Models & Runtimes**
- Leverage new features in MPICH, OpenMP libraries
- Use C++ compile-time polymorphism to generate node-specific code from common source code (e.g., Kokkos, RAJA)
- Experiment with alternative programming models (Legion, UPC++/GASNet)

**Development Tools**
- Tools for performance analysis:
  - PAPI, TAU, HPC Toolkit, Dyninst
  - Widely used in HPC community
- Portable, open-source LLVM compiler ecosystem to support expected ECP architectures, including support for F18

**Math Libraries**
- Use hypre, PETSc, SuperLU, Trilinos, others: All widely used parallel solvers being adapted for massive on-node concurrency.
  - APIs are largely unchanged
  - Provides performance portability across platforms
  - Try STRUMPACK
  - Suitable SuperLU replacement
  - Highly scalable (for a direct solver).
  - Turnkey solver (easy to install and use)

**Data and Visualization**
- New storage software and workflows associated with non-volatile memory
  - Fundamental I/O game-changer
  - Examples: Fast offload of checkpoints, all-flash storage system
- Data compression tools: Same impact as increasing memory and storage size and bandwidth.
  - In situ workflows: Increased opportunities to analyze and transform data as part of the workflow.

**Software Ecosystem**
- Advanced resource management:
  - Fast, scalable checkpoint/restart (leverage NVRAM).
  - Resource managers, e.g., Flux.
- SDKs and Spack are emerging as attractive combination for managing software components:
  - Involvement and input from industry can be beneficial both ways
## Some ECP-Industry Collaboration Models

<table>
<thead>
<tr>
<th>Approach</th>
<th>Comments/Potential</th>
</tr>
</thead>
<tbody>
<tr>
<td>Read ECP-related papers</td>
<td>Traditional Approach. Works well for small scope: algorithmic advances.</td>
</tr>
<tr>
<td>Attend ECP-related tutorials and webinars</td>
<td>Many ST technologies offer tutorial/webex forums to learn more; range from introductory to advanced</td>
</tr>
<tr>
<td>Develop <em>de facto</em> and ISO standards</td>
<td>MPI, OpenMP, C++, Fortran, PAPI, BLAS: Happening, more is better.</td>
</tr>
<tr>
<td>Evaluate/prototype new capabilities using ECP software products</td>
<td>Kokkos, STRUMPACK/SuperLU and more: Prototyping and proof-of-concept is a success story, especially if giving feedback from experience.</td>
</tr>
<tr>
<td>Adopt and rely upon ECP software (as an option)</td>
<td>A goal for us: Want to explore how to make this possible. Collaboration can help us improve our product development and delivery.</td>
</tr>
<tr>
<td>Software Engineering practices</td>
<td>ECP raises expectations on DOE software. Collaboration with industry can accelerate our progress.</td>
</tr>
<tr>
<td>Overall</td>
<td>Two way interactions allow ECP to help industry and industry to help ECP.</td>
</tr>
</tbody>
</table>
The ECP is on track to deliver a capable exascale computing ecosystem

**Applications**
- 25 application teams actively engaged in targeted development and capability enablement for 2+ years
- Apps have well-defined exascale challenge problem targets with associated “science work rate” goals
- Initial performance experiences on pre-exascale systems (Summit, Sierra) exceeding expectations

**Software Stack**
- Over 80 software technology products being actively developed for next generation architectures
- Regular assessment of software stack products ensures line-of-sight to apps and HPC Facilities
- Plans for broad containerized delivery of products via SDKs and the E4S being executed

**Hardware & Integration**
- Return on PathForward vendor hardware R&D element evident in recent exascale RFP responses
- Plans for deployment and continuous integration of SDKs into DOE HPC Facilities being executed
- Prioritized performance engineering of applications targeting first three exascale systems underway
For more information…

https://www.exascaleproject.org

or reach out to the leadership team in the areas that interest you..

Doug Kothe, ECP Director
do@ornl.gov

Lori Diachin, ECP Deputy Director
lora2@llnl.gov

Andrew Siegel, Application Development Director
seigela@uchicago.edu

Erik Draeger, Application Development Deputy Director
draeger1@llnl.gov

Mike Heroux, Software Technology Director
maherou@sandia.gov

Jonathan Carter, Software Technology Deputy Director
jtcarter@lbl.gov

Terri Quinn, Hardware and Integration Director
quinn1@llnl.gov

Susan Coghlan, Hardware and Integration Deputy Director
smc@alcf.anl.gov
Stay Informed of ECP Activities

Go to the ECP homepage

Exascaleproject.org

Simply enter your email and you will be subscribed to ECP newsletters and frequent updates